Search Results

Search Results for: **[(data <near/2> (stream or strip)) <paragraph> column<AND>((((parsing or partition) <near/9> column)))]**
Found **8** of **127,132 searched.**

Search within Results

GO > Advanced Search

> Search Help/Tips

Sort by: **Title** **Publication** **Publication Date** **Score** Binder

Results 1 - 8 of 8 **short listing**

1 Set-oriented production rules in relational database systems 100%

Jennifer Widom , S. J. Finkelstein

ACM SIGMOD Record , Proceedings of the 1990 ACM SIGMOD international conference on Management of data May 1990

Volume 19 Issue 2

We propose incorporating a production rules facility into a relational database system. Such a facility allows definition of database operations that are automatically executed whenever certain conditions are met. In keeping with the set-oriented approach of relational data manipulation languages, our production rules are also set-oriented—they are triggered by sets of changes to the database and may perform sets of changes. The condition and action parts of our production rules may r ...

2 Pipelined data parallel algorithms—concept and modeling 100%

C.-T. King , W.-H. Chou , L. M. Ni

Proceedings of the 2nd international conference on Supercomputing June 1988

A new style of efficient parallel algorithms on distributed-memory multiprocessors is introduced, which exploits parallelism through pipelined parallel computation, or large-grain pipelining. By using macro-pipelining between nodes in the system, large-grain pipelining regulates the flows of data in the multiprocessor so that the degree of overlapping can be maximized and the effect of communication overhead can be minimized. To model pipelined parallel computations, an ana ...

3 Code extension in ASCII 100%

S. Gorn

Communications of the ACM October 1966

Volume 9 Issue 10

The American Standard Code for Information Interchange (ASCII) contains a number of control characters associated with the principle of code extension, that is, with the representation of information which cannot be directly represented by means of the

characters in the Code. The manner of use of these characters has not previously been completely described. This paper presents a set of mutually consistent philosophies regarding code extension applications, and s ...

4 Comparison of two-dimensional FFT methods on the hypercube 99%

C. Y. Chu

Proceedings of the third conference on Hypercube concurrent computers and applications - Volume 2 January 1989

Complex two-dimensional FFTs up to size 256 x 256 points are implemented on the Intel iPSC/System 286 hypercube with emphasis on comparing the effects of data mapping, data transposition or communication needs, and the use of distributed FFTs. Two new implementations of the 2D-FFT include the Local-Distributed method which performs local FFTs in one direction followed by distributed FFTs in the other direction, and a Vector-Radix implementation that is derived from decimating the DFT in two ...

5 A modular systolic architecture for image convolutions 99%

K. Doshi , P. Varman

Proceedings of the 14th annual international symposium on Computer architecture June 1987

This paper describes a modular, systolic design for two-dimensional convolution which is a frequent and computationally intensive operation in low-level image processing. The design consists of a one-dimensional array of homogeneous cells, each with a fixed amount of storage. The paper also presents schema by which the design consisting of a limited number of cells can be used to implement convolutions of varying kernel siz ...

6 Application-specific memory management for embedded systems using software-controlled caches 99%

Derek Chiou , Prabhat Jain , Larry Rudolph , Srinivas Devadas

Proceedings of the 37th conference on Design automation June 2000

We propose a way to improve the performance of embedded processors running data-intensive applications by allowing software to allocate on-chip memory on an application-specific basis. On-chip memory in the form of cache can be made to act like scratch-pad memory via a novel hardware mechanism, which we call column caching. Column caching enables dynamic cache partitioning in software, by mapping data regions to a specified sets of cache "columns" or "ways ...

7 Dynamic object management for distributed data structures 98%

B. K. Totty , D. A. Reed

Proceedings of the 1992 ACM/IEEE conference on Supercomputing December 1992

8 An overview of query optimization in relational systems 97%

Surajit Chaudhuri

Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems May 1998

Results 1 - 8 of 8 short listing

h c g e cf c

Inc.

h c g e cf c

Search Results

Search Results for: **[(data <near/2> (stream or strip)) <near/9> column<AND> ((((parsing or partition) <near/29> column)))]**
Found **3** of **127,132 searched.**

Search within Results

GO > Advanced Search

> Search Help/Tips

Sort by: **Title** **Publication** **Publication Date** **Score** Binder

Results 1 - 3 of 3 **short listing**

1 Comparison of two-dimensional FFT methods on the hypercube 98%
C. Y. Chu
Proceedings of the third conference on Hypercube concurrent computers and applications - Volume 2 January 1989
Complex two-dimensional FFTs up to size 256 x 256 points are implemented on the Intel iPSC/System 286 hypercube with emphasis on comparing the effects of data mapping, data transposition or communication needs, and the use of distributed FFTs. Two new implementations of the 2D-FFT include the Local-Distributed method which performs local FFTs in one direction followed by distributed FFTs in the other direction, and a Vector-Radix implementation that is derived from decimating the DFT in two ...

2 An overview of query optimization in relational systems 97%
Surajit Chaudhuri
Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems May 1998

3 Pipelined data parallel algorithms—concept and modeling 95%
C.-T. King , W.-H. Chou , L. M. Ni
Proceedings of the 2nd international conference on Supercomputing June 1988
A new style of efficient parallel algorithms on distributed-memory multiprocessors is introduced, which exploits parallelism through pipelined parallel computation, or large-grain pipelining. By using macro-pipelining between nodes in the system, large-grain pipelining regulates the flows of data in the multiprocessor so that the degree of overlapping can be maximized and the effect of communication overhead can be minimized. To model pipelined parallel computations, an ana ...

Results 1 - 3 of 3 **short listing**

h c g e cf c

Search Results

Search Results for: **[(data <near/2> (stream or strip))<AND>((((parsing or partition) <near/2> column)))]**
Found **10** of **127,132 searched.**

Search within Results

GO > Advanced Search

> Search Help/Tips

Sort by: **Title** **Publication** **Publication Date** **Score** Binder

Results 1 - 10 of 10 **short listing**

1 Large-grain pipelining on hypercube multiprocessors 100%

C-T. King , L. M. Ni

Proceedings of the third conference on Hypercube concurrent computers and applications - Volume 2 January 1989

A new paradigm, called large-grain pipelining, for developing efficient parallel algorithms on distributed-memory multiprocessors, e.g., hypercube machines, is introduced. Large-grain pipelining attempts to maximize the degree of overlapping and minimize the effect of communication overhead in a multiprocessor system through macro-pipelining between the nodes. Algorithms developed through large-grain pipelining to perform matrix multiplication are presented. To model the pi ...

2 Warp experience: we can map computations onto a parallel computer efficiently 100%

H. T. Kung

Proceedings of the 2nd international conference on Supercomputing June 1988

Warp is a programmable, systolic array computer developed by Carnegie Mellon and produced by GE. A 10-cell Warp machine can perform 100 million floating-point operations per second (10 MFLOPS). A variety of applications have been mapped onto Warp. The experience has been that the mapping is not a real problem; in fact, usually near-optimal mapping is relatively easy to obtain, and the actual implementation of the mapping on the machine can often be automated. This paper explains why this is ...

3 Pipelined data parallel algorithms—concept and modeling 100%

C.-T. King , W.-H. Chou , L. M. Ni

Proceedings of the 2nd international conference on Supercomputing June 1988

A new style of efficient parallel algorithms on distributed-memory multiprocessors is introduced, which exploits parallelism through pipelined parallel computation, or large-grain pipelining. By using macro-pipelining between nodes in the system, large-grain pipelining regulates the flows of data in the multiprocessor so that the

degree of overlapping can be maximized and the effect of communication overhead can be minimized. To model pipelined parallel computations, an ana ...

4 Automatic data layout for distributed-memory machines 100%

Ken Kennedy , Ulrich Kremer

ACM Transactions on Programming Languages and Systems (TOPLAS) July 1998 Volume 20 Issue 4

The goal of languages like Fortran D or High Performance Fortran (HPF) is to provide a simple yet efficient machine-independent parallel programming model. After the algorithm selection, the data layout choice is the key intellectual challenge in writing an efficient program in such languages. The performance of a data layout depends on the target compilation system, the target machine, the problem size, and the number of available processors. This makes the choice of a good layout extremel ...

5 Compiler optimizations for Fortran D on MIMD distributed-memory machines 100%

Seema Hiranandani , Ken Kennedy , Chau-Wen Tseng

Proceedings of the 1991 ACM/IEEE conference on Supercomputing August 1991

6 Set-oriented production rules in relational database systems 99%

Jennifer Widom , S. J. Finkelstein

ACM SIGMOD Record , Proceedings of the 1990 ACM SIGMOD international conference on Management of data May 1990 Volume 19 Issue 2

We propose incorporating a production rules facility into a relational database system. Such a facility allows definition of database operations that are automatically executed whenever certain conditions are met. In keeping with the set-oriented approach of relational data manipulation languages, our production rules are also set-oriented—they are triggered by sets of changes to the database and may perform sets of changes. The condition and action parts of our production rules may r ...

7 Comparison of two-dimensional FFT methods on the hypercube 99%

C. Y. Chu

Proceedings of the third conference on Hypercube concurrent computers and applications - Volume 2 January 1989

Complex two-dimensional FFTs up to size 256 x 256 points are implemented on the Intel iPSC/System 286 hypercube with emphasis on comparing the effects of data mapping, data transposition or communication needs, and the use of distributed FFTs. Two new implementations of the 2D-FFT include the Local-Distributed method which performs local FFTs in one direction followed by distributed FFTs in the other direction, and a Vector-Radix implementation that is derived from decimating the DFT in two ...

8 Code extension in ASCII 99%

S. Gorn

Communications of the ACM October 1966 Volume 9 Issue 10

The American Standard Code for Information Interchange (ASCII) contains a number of control characters associated with the principle of code extension, that is, with the representation of information which cannot be directly represented by means of the characters in the Code. The manner of use of these characters has not

previously been completely described. This paper presents a set of mutually consistent philosophies regarding code extension applications, and s ...

9 A modular systolic architecture for image convolutions 99%

K. Doshi , P. Varman

Proceedings of the 14th annual international symposium on Computer architecture June 1987

This paper describes a modular, systolic design for two-dimensional convolution which is a frequent and computationally intensive operation in low-level image processing. The design consists of a one-dimensional array of homogeneous cells, each with a fixed amount of storage. The paper also presents schema by which the design consisting of a limited number of cells can be used to implement convolutions of varying kernel siz ...

10 Solving narrow banded systems on ensemble architectures 99%

S. Lennart Johnsson

ACM Transactions on Mathematical Software (TOMS) September 1985 Volume 11 Issue 3

We present concurrent algorithms for the solution of narrow banded systems on ensemble architectures, and analyze the communication and arithmetic complexities of the algorithms. The algorithms consist of three phases. In phase 1, a block tridiagonal system of reduced size is produced through largely local operations. Diagonal dominance is preserved. If the original system is positive, definite, and symmetric, so is the reduced system. It is solved in a second phase, and the remaining varia ...

Results 1 - 10 of 10 **short listing**

Your search matched **14** of **1003743** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

(parsing or partitioning) <near/5> columns Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

1 **Mapping the Kalman tracking algorithm onto the transputer networ**
Vaidehi, V.; Krishnan, C.N.;
Aerospace and Electronic Systems, IEEE Transactions on , Volume: 34 , Issue 2 , April 1998
Pages:681 - 686

[Abstract] [PDF Full-Text (76 KB)] **IEEE JNL**

---

2 **Data partitioning schemes for the parallel implementation of the re simplex algorithm for LP problems**
Sridhar, U.; Basu, A.;
Parallel Processing Symposium, 1993., Proceedings of Seventh International , 16 April 1993
Pages:379 - 383

[Abstract] [PDF Full-Text (376 KB)] **IEEE CNF**

---

3 **Combining logic minimization and folding for PLAs**
Yu Chin Hsu; Youn Long Lin; Hang Ching Hsieh; Ting Hai Chao;
Computers, IEEE Transactions on , Volume: 40 , Issue: 6 , June 1991
Pages:706 - 713

[Abstract] [PDF Full-Text (592 KB)] **IEEE JNL**

---

4 **Novel fault-tolerant techniques for high capacity RAMs**
Chih-Hsien Hsu; Shyue-Kung Lu; Sy-Yen Kuo;
Dependable Computing, 2001. Proceedings. 2001 Pacific Rim International Symposium on , 17-19 Dec. 2001
Pages:11 - 18

h eee e eee g e ch e ch e e e g g

[Abstract] [PDF Full-Text (844 KB)] IEEE CNF

5 **Hierarchical fault tolerance for 3D microelectronics**
Campbell, M.; Little, M.; Yung, M.;
Wafer Scale Integration, 1990. Proceedings., [2nd] International Conference on , 23-25 Jan. 1990
Pages:174 - 188

[Abstract] [PDF Full-Text (840 KB)] IEEE CNF

6 **Processing reporting function views in a data warehouse environme**
Lehner, W.; Hummer, W.; Schlesinger, L.;
Data Engineering, 2002. Proceedings. 18th International Conference on , 26 I March 2002
Pages:176 - 185

[Abstract] [PDF Full-Text (463 KB)] IEEE CNF

7 **Reduced-order realization of a nonlinear power network using companion-form state equations with periodic coefficients**
Noda, T.; Semlyen, A.; Iravani, R.;
Power Delivery, IEEE Transactions on , Volume: 18 , Issue: 4 , Oct. 2003
Pages:1478 - 1488

[Abstract] [PDF Full-Text (643 KB)] IEEE JNL

8 **A register file with transposed access mode**
Yoochang Jung; Berg, S.G.; Donglok Kim; Yongmin Kim;
Computer Design, 2000. Proceedings. 2000 International Conference on , 17-Sept. 2000
Pages:559 - 560

[Abstract] [PDF Full-Text (148 KB)] IEEE CNF

9 **New partitioning schemes for parallel modified Gram-Schmidt orthogonalization**
Oliveria, S.; Soma, T.;
Parallel Architectures, Algorithms, and Networks, 1997. (I-SPAN '97) Proceed Third International Symposium on , 18-20 Dec. 1997
Pages:233 - 239

[Abstract] [PDF Full-Text (512 KB)] IEEE CNF

10 **Fast parallel algorithms for solving triangular systems of linear equations on the hypercube**
Ibarra, O.H.; Kim, M.H.;
Parallel Processing Symposium, 1991. Proceedings., Fifth International , 30 A May 1991
Pages:76 - 83

[Abstract] [PDF Full-Text (580 KB)] IEEE CNF

11 **A 30M samples/s programmable filter processor**

Golla, C.; Nava, F.; Cavallotti, F.; Cremonesi, A.; Piacentini, P.; Casagrande, Campardo, G.;
Solid-State Circuits Conference, 1990. Digest of Technical Papers. 37th ISSC(1990 IEEE International , 14-16 Feb. 1990
Pages:117 - 117, 276

[Abstract] [PDF Full-Text (636 KB)] **IEEE CNF**

---

12 **An efficient VLSI switch-box router**
Jer Min Jou; Jau Yien Lee; Yachyang Sun; Jhing Fa Wang;
Design & Test of Computers, IEEE , Volume: 7 , Issue: 4 , Aug. 1990
Pages:52 - 65

[Abstract] [PDF Full-Text (796 KB)] **IEEE JNL**

---

13 **Yield analysis for repairable embedded memories**
Sehgal, A.; Dubey, A.; Marinissen, E.J.; Wouters, C.; Vranken, H.; Chakrabar K.;
European Test Workshop, 2003. Proceedings. The Eighth IEEE , 25-28 May 2(Pages:35 - 40

[Abstract] [PDF Full-Text (345 KB)] **IEEE CNF**

---

14 **Efficient solution of GSPNs using canonical matrix diagrams**
Miner, A.S.;
Petri Nets and Performance Models, 2001. Proceedings. 9th International Wor on , 11-14 Sept. 2001
Pages:101 - 110

[Abstract] [PDF Full-Text (800 KB)] **IEEE CNF**

---

h eee e eee g e ch e ch e e e g g

Your search matched **1** of **1003743** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enter new one in the text box.

(column <near/2> files) Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

1 **Engineering baselines in system development: using ASCII files, tw column index piles, and system numbers, engineering tags, and chan set numbers**
Evans, R.P.; Sooyong Park; Merriman, M.;
Engineering of Complex Computer Systems, 1995. Held jointly with 5th CSES 3rd IEEE RTAW and 20th IFAC/IFIP WRTP, Proceedings., First IEEE Internatio Conference on , 6-10 Nov. 1995
Pages:17 - 20

[Abstract] [PDF Full-Text (400 KB)] **IEEE CNF**

h eee e eee g e ch e ch e e e c e

Search Results

Search Results for: **[(parsing or partitioning) <near/9> column<AND>((column <near/2> files))]**
Found **6** of **127,132 searched.**

Search within Results

GO > Advanced Search

> Search Help/Tips

Sort by: **Title** **Publication** **Publication Date** **Score** Binder

Results 1 - 6 of 6 **short listing**

1 GEMM-based level 3 BLAS: high-performance model implementations and performance evaluation benchmark 99%

Bo Kågström , Per Ling , Charles van Loan
ACM Transactions on Mathematical Software (TOMS) September 1998
Volume 24 Issue 3

The level 3 Basic Linear Algebra Subprograms (BLAS) are designed to perform various matrix multiply and triangular system solving computations. Due to the complex hardware organization of advanced computer architectures the development of optimal level 3 BLAS code is costly and time consuming. However, it is possible to develop a portable and high-performance level 3 BLAS library mainly relying on a highly optimized GEMM, the routine for the general matrix multiply and add operation. With s ...

2 Full-screen, scrollable APL2 spreadsheet input/output editor 99%

Peter A. W. Lewis
ACM SIGAPL APL Quote Quad March 1993
Volume 23 Issue 3

A full-screen, scrollable, spreadsheet-like editor based on IBM's APL2 32-bit interpreter for 386/486-based microcomputers is described. It is used for entering, examining, analyzing, editing and printing data. Mixed numeric and character arrays can be read in from or written out to formatted DOS files (ASCII) or comma-delimited DOS files. Alternatively, a bulk mode input facility allows for rapid direct data entry, or data can be entered, examined and edited cell-by-cell in the usual way. A fac ...

3 Extensible file system (ELFS): an object-oriented approach to high performance file I/O 97%

John F. Karpovich , Andrew S. Grimshaw , James C. French
ACM SIGPLAN Notices , Proceedings of the ninth annual conference on Object-oriented programming systems, language, and applications October 1994

Volume 29 Issue 10

Scientific applications often manipulate very large sets of persistent data. Over the past decade, advances in disk storage device performance have consistently been outpaced by advances in the performance of the rest of the computer system. As a result, many scientific applications have become I/O-bound, i.e. their run-times are dominated by the time spent performing I/O operations. Consequently, the performance of I/O operations has become critical for high performance in these applicatio ...

4 The architecture of Montana: an open and extensible programming environment with an incremental C++ compiler 96%

Michael Karasick

ACM SIGSOFT Software Engineering Notes , Proceedings of the 6th ACM SIGSOFT international symposium on Foundations of software engineering November 1998

Volume 23 Issue 6

Montana is an open, extensible integrated programming environment for C++ that supports incremental compilation and linking, a persistent code cache called a CodeStore, and a set of programming interfaces to the CodeStore for tool writers. CodeStore serves as a central source of information for compiling, browsing, and debugging. CodeStore contains information about both the static and dynamic structure of the compiled program. This information spans files, macros, declarations, function bodies, ...

5 A Functional Description of ANALYZE: A Computer-Assisted Analysis System for Linear Programming Models 95%

Harvey Greenberg

ACM Transactions on Mathematical Software (TOMS) March 1983

Volume 9 Issue 1

6 Automating the re-declaration of unneeded globals as private 95%

Amitava Datta , Prabhaker Mateti

Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing: states of the art and practice March 1993

Results 1 - 6 of 6 short listing

---

h c g e cf c